EasyChair Preprint

№ 8789

Recognition of Aquatic Invasive Species Larvae Using Autoencoder-based Feature Averaging

Shaif Chowdhury and Greg Hamerly

September 5, 2022

Recognition of Aquatic Invasive Species Larvae using Autoencoder-based Feature Averaging

Shaif Chowdhury and Greg Hamerly

Baylor University, Waco, TX 76798, USA

Abstract. The spread of invasive aquatic species disrupts ecological balance, damages natural resources, and adversely affects agricultural activity. There is a need for automated systems that can detect, track and classify invasive and non-invasive aquatic species using underwater videos without human supervision. In this paper, we intend to classify the larvae (aka veligers) of invasive species like Zebra and Quagga mussels. These organisms are native to eastern Europe, but are invasive in United States waterways. It's important to identify invasive species at the larval stage when they are mobile in the water and before they have established a presence, to avoid infestations. Video-based underwater species classification has several challenges due to variation of illumination, angle of view and background noise. In the case of invasive larvae, there is added difficulty due to the microscopic size and small differences between aquatic species larvae. Additionally, there are challenges of data imbalance since invasive species are typically less abundant than native species. In video-based surveillance methods, each organism may have multiple video frames offering different views that show different angles, conditions, etc. Because there are multiple images per organism, we propose using image set based classification which can accurately classify invasive and non-invasive organisms based on sets of images. Image-set classification can often have higher accuracy even if single image classification accuracy is lower. Our proposed system classifies image-sets with a feature averaging pipeline that begins with an autoencoder to extract features from the images. These features are then averaged for each set. In our case, each set corresponds to a single organism. The final prediction is made by a classifier trained on the image set features. Our experiments show that feature averaging provides a significant improvement over other models of image classification, achieving more than 97% F1 score to predict invasive organisms on our video imaging data for a quagga mussel survey.

Keywords: Invasive Species · Quagga mussels · Classification · Image set · Autoencoder · Feature Averaging.

1 Introduction

Zebra mussels (*Dreissena polymorpha*) and Quagga mussels (*Dreissena bugensis*) are not native to North American waters and probably arrived as freshwater stowaways in commercial vessels from Europe in the 1980s [37]. Zebra mussels spread rapidly, cause ecological disruption, and clog water pipes and other machinery [11, 34]. Due to economical and environmental need it is important to detect and prevent the spread of these invasive species. Adult zebra mussels are easy to spot but they can spread quickly by laying millions of eggs per season. By the time these invasive species have established themselves in a waterway, eradicating or mitigating their presence becomes very difficult and costly. That means, it is important to detect and monitor zebra mussels at the larval (aka veliger) stage [26] to stop the spread in waterways. Traditionally, detection of veligers is usually done by collecting water samples and then using microscopy with cross-polarized light for identification [26], or using DNA-based methods [12]. Microscopy is very expensive and time-consuming, and requires experts to check the water samples manually. DNA-based methods are also time-consuming and expensive, and are able to detect the presence but not *prevalence*. This is why it is necessary to have an automated process to visually monitor both veliger presence and prevalence [15].

Recently, there has been a lot of research in classifying fish and other underwater species [4, 45, 51]. But, there can be some unique challenges in classifying veligers of invasive species. First, fish and other adult underwater species have large and recognizable patterns, while veligers are difficult to distinguish from other organisms even for human experts. Secondly, there are a lot of other native planktonic organisms present in the water samples. Moreover, veligers can be rare depending on the season, which creates a data imbalance problem both at the training and testing stage [26]. Additionally, images collected from water samples might

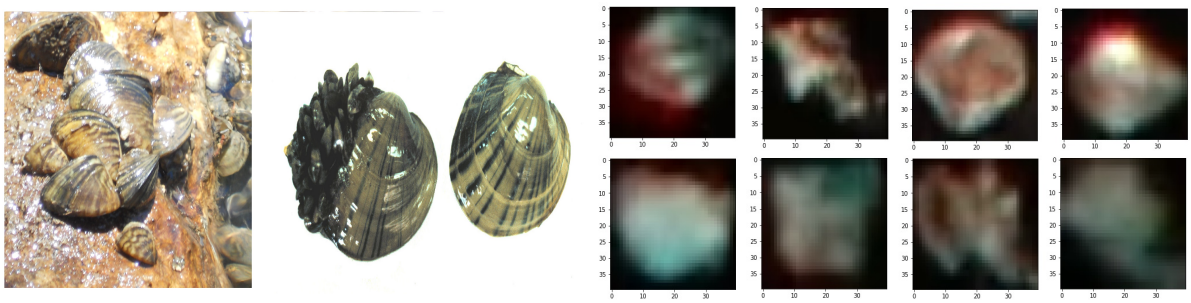


Fig. 1. 1. Adult zebra mussels that are easily recognizable. But our problem is about detecting zebra mussels at larval (veliger) stage, so the spread of invasive species can be stopped. 2. Some images of veligers from our dataset. The first row contains images of invasive veligers, and the second row contains images of other non-invasive organisms.

vary in illumination and background noise, and viewpoint orientation. Therefore, any solution for detecting invasive veligers must take the aforementioned challenges into account.

Our dataset comes from a video capture of a water sample. First, organisms in the video are tracked and then cropped images are extracted for each organism in the video frame. This is done by proprietary software developed by a private company, and is based on a Kalman filter. For each tracked organism, we group together its extracted images, and aim to classify the set as either invasive or non-invasive. That means our prediction model is based on image set classification, i.e. classification based on multiple images of same object [29]. The ground truth is provided by experts inspecting the tracked objects on the video as well as the cropped images.

Image set based classification is often used in face detection with multiple instances of the same person recorded from surveillance videos. These datasets generally contain images of faces captured under different poses, expressions, or illumination [28,42]. An image-set based approach can perform better than single image classification, given that they take advantage of the multiple instances available [42,47]. The general solution to this problem entails reducing the dimension of the images, followed by aggregation of features of images in the same set. The second step is to use a similarity or distance measure with a nearest neighbour classifier [41, 54]. This is both computationally expensive and unreliable for images with fine-grained differences.

Our dataset has two primary classes, which we call invasive and non-invasive. Each organism has multiple extracted images, where the number of extracted images varies depending on how long the organism was in the video frame. Our solution is based on a feature extraction model followed by a final classifier. For feature extraction we have considered both hand-crafted and deep learning based methods. Hand-crafted features generally use a filter to encode some characteristics of an image like edges, color, shape, etc. Some popular hand-crafted feature descriptors are SIFT, HOG, HSV color histogram, PCA, etc. [31]. More recently, deep learning based methods are able to model complex image features much more accurately [14]. So, for this work we used a convolutional autoencoder to extract features from individual images [32, 43].

In the last decade, there has been a lot of growth in deep learning methods for machine learning tasks. In particular, convolutional neural networks have been able to achieve significant improvement in many learning tasks, especially image classification. In particular, non-linear activation functions, batch normalization, pooling, and regularization layers have improved network performance [7]. In our case, we use a convolution autoencoder to map images to lower-dimensional features. Autoencoders use an encoder to create latent representation from an image and then a decoder is used to reconstruct the original image [5]. The loss is calculated based on the difference between original and reconstructed image and is optimized over the training period. Since the latent representation is created over multiple layers, it presents an opportunity to use different activation functions and create features that are appropriate for the problem.

Another challenge in our invasive species dataset is that the images are taken in different angles and illuminations. This is clearly shown in Fig2. with groups of invasive and non-invasive images of the same organism placed side by side. The variations come from the organisms moving in three dimensions as they pass through the video frame. Thus we need a machine learning pipeline that is invariant to different conditions [3]. One technique is to train different models for different purposes and then use an ensemble for the final prediction. This is of course more expensive to train and might also introduce bias towards certain types of

data variations. For example, the ensemble model might over-fit on images of low illumination and do poorly against images of high illumination. Another technique is to augment the dataset, which aims to create data variations resulting in a more balanced dataset [44]. This approach can involve changing brightness of some images with low illumination and using them to balance the dataset [22]. There are quite a few papers that propose using morphological transformation or even using generative adversarial network to create augmented data samples [48]. These ideas are relevant to our problem, especially since we also have data imbalance in favor of non-invasive species. But generative adversarial networks with classification models are difficult to train, risk overfitting, and are computationally expensive.

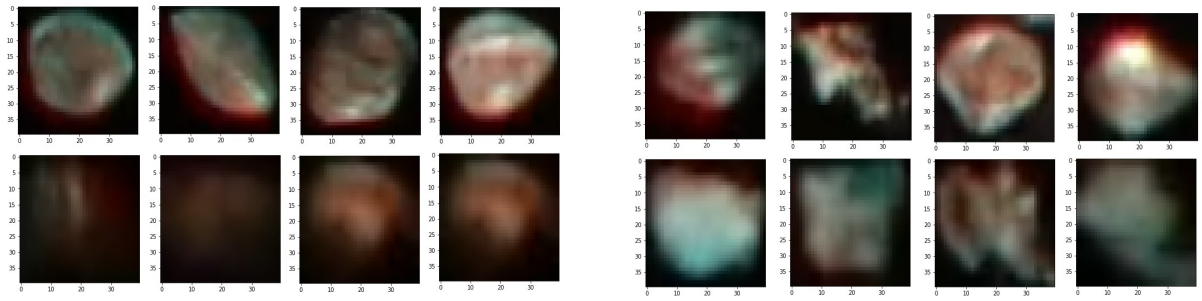


Fig. 2. These are groups of four images placed in same row taken from the same organism. As shown here, the images have a lot of variation in terms of viewpoint and illumination, which makes it difficult to accurately classify individual samples.

In this paper, we present a feature averaging process to create a representation from an image set which makes classification robust to varying illumination and object orientation and also reduces generalization error [28]. The key idea is that using the average of multiple images reduces the effect of illumination or viewpoint changes, compared to the use of a single image [27]. There are papers that use linear or affine subspace based methods to represent mean or basis image from image sets [41]. But, we have decided to use element-wise average of autoencoder features, which is a more robust representation of the image set, while capturing the fine details at the same time.

Our classification pipeline is based on two neural networks. At first an autoencoder is used to extract features from each instance of an organism. Then these features are averaged across images of the same organism to create a single feature vector for the organism. Finally, we use a classifier to predict if the organism is invasive or non-invasive. Our main contribution here is to present an approach of classification that can take advantage of multiple instances and can accurately, reliably classify invasive and non-invasive larvae despite the presence of variation in illumination and viewpoint. Our experiments also show the robustness of autoencoder based feature-averaging process compared to other classification models. The balanced accuracy of this method is 97% on the test data, which is a significant improvement from other previous models, that were also convolutional neural networks similar to VGGNet [46]. In the results section, we provide more detailed results including F1-score, recall, and balanced accuracy. We also compare our results with other proposed methods of classifying underwater images or image sets like CNN, PCA+CNN, SVM etc.

The rest of the paper is structured as follows: in the next section we provide a literature survey of this problem including previous research on invasive species detection, local responses to the spread of invasive aquatics Species, image set-based classification methods, underwater image classification, and neural network models for feature extraction. In the methodology section we discuss the detailed architecture of our classification pipeline, network structure, loss functions for training, and the dataset. The results section gives details about the choice of evaluation metrics and finally presents and analyzes the results. In the conclusion we discuss the main contribution of the paper and present plans for future work.

2 Related Work

The problem of aquatic invasive species is not new, but approaches for detection and identification of zebra mussels are costly and can be ineffective at preventing infestation due to the time required. Common approaches for detecting invasive mussels include microscopy or environmental DNA (eDNA). Here we review some recent approaches for detection of invasive species as well as some action plans to stop their further spread.

2.1 Invasive Aquatic Species

Most of the early techniques for detection of zebra mussels in larval stage are based on microscope photography. Conn et al. [10] provide a framework to detect and differentiate between larval and post-larval stages of zebra mussel (Dreissena polymorpha) and the Dark False mussel (Mytilopsis leucophaeata). This photographic guide aims to help personnel involved with the monitoring of these organisms.

The other notable work in this area is from Johnson et al. [26], which uses cross-polarizing filters for microscopy retrofitted to detect the presence of zebra mussel veligers much faster with improved accuracy. This technique is useful for rapid detection as well as counting of veligers in a water sample.

A recent study by Gingera et al. [20] is based on water samples from Lake Winnipeg during early May and late October. This is an eDNA-based technique to identify the presence of zebra and quagga mussels. The results of the study show that zebra mussels were detected in $0 - 33.3\%$ of all water samples per site studied during the early season and $42.9 - 100\%$ during the late season.

Marshall et al. [33] presented another eDNA-based approach to detect and distinguish between invasive species zebra and quagga mussels. This is based on field collected water from the Great Lakes (Lake Erie) and the Hudson River.

Finally, Feist et al. [17] provides a detailed review of eDNA-based approaches to detect and combat the spread of zebra and quagga mussels along with discussion on all the important discoveries and novel revelations made along the way.

Other than detection of invasive species, there is a lot of research on how to trace and combat the presence of zebra and quagga mussels. The Massachusetts Department of Conservation and Recreation [35] has a rapid response plan to combat spread of zebra mussels, which involves collection of water samples, early detection of invasive species, marking of GPS position for infested locations followed by risk assessment and necessary response. Automated detection and classification of invasive species is crucial for early detection and response to the growth of invasive species.

2.2 Local Responses to Invasive Aquatic Species

Now we look at some of the different states across western United States and ways they monitor and control aquatic invasive species. In the state of Texas freshwater fisheries and other aquatic resources are managed by the Texas Parks and Wildlife Department (TPWD). Experts from TPWD and their partner organizations monitor Texas water bodies for the spread of zebra and quagga mussels twice a year – once every Fall and Spring. There are different amounts of infestation in different lakes. For example, Lake Worth in Tarrant County, Lake Brownwood, Inks Lake, and Medina Lake in the Colorado and San Antonio River basins have been designated as infested which indicates a sustained significant presence of zebra mussels in those lakes. On the other hand, International Amistad Reservoir in the Rio Grande basin had the first detection of quagga mussels in a Texas reservoir in February 2022.

In California spread of quagga mussels have happened in Southern California reservoirs that receive water from the Colorado River. The state of California had added legislation requiring all reservoir owners and managers to assess the possibility of zebra and quagga mussels spread [19].

In the state of Arizona there has been early detection of zebra mussels. The Arizona Game and Fish Department (AZGFD) has urged pet stores and aquarium owners to check for possible infestation of zebra mussels [1].

Overall, the spread of invasive species is estimated to have an economic impact of $219 billion in United States affecting different types of water infrastructure along with fishing, boating, hunting etc. Worldwide it is estimated to have economic impact of more than $4 trillion. This makes automated early detection and monitoring of zebra and quagga mussels crucial to reduce the environmental as well economic damage [49].

2.3 Image-set Based Classification Methods

Our dataset contains multiple images of the same organism taken from video sample with different pose and illumination. Because of that, we chose an approach based on image set classification. Over the years there has been a lot of interest in image set based classification [18, 29, 55], especially in the area of face recognition [2, 8, 16, 23], hand-written digit recognition [24], shape recognition [13] and object recognition from different viewpoints [29]. The general procedure for image set classification is as follows: images of the same class are grouped together. A model or a probability distribution is used to represent the set. Now for test data a similarity measurement is used to match the set with a particular class. So, the key problem of image set classification is to capture the intrinsic properties of the set and use those for classification.

Most image set classification approaches can be categorized into two different types: parametric models and non-parametric models [55]. Parametric models assume that each population follows a certain distribution, determined by parameters. In this method each image set is modelled using a distribution function and a similarity measure is used make the final classification.

Non-parametric methods do not rely on the statistical correlation between the training set and the distribution fitting of samples. These methods create a representation for each image set and then a distance measure is used for the prediction. These methods represent image sets in a number of different ways [55] like: linear subspace methods [29, 53], nonlinear manifold methods [2, 16, 18, 52], and affine subspace methods [6].

Linear subspace methods place images in a low dimensional linear subspace and use the subspace distance as a measure of similarity. For distance measure, Euclidean distance and cosine distance are among the most frequently used. Yamaguchi et al. [53] represented face images from different direction and expression to create a subspace with the image sequence and apply the Mutual Subspace Method as a distance measure. Kim et al. [29] developed a discriminative model, which maximizes the canonical correlations within sets of same class. This method has been evaluated on various image set datasets, including Cambridge-Toshiba Video-based Face Database, ETH80 object recognition dataset.

The nonlinear manifold method represents images from the same set as a nonlinear manifold. Wang et al. propose a manifold learning approach [52], expressing each manifold by a collection of linear models. Image sets from the test data are mapped to the manifold and matched against manifolds from the training set. The final classification is made by calculating the Manifold-Manifold Distance (MMD).

Cevikalp et al. [6] proposed an affine subspace based method, where images from the same set are presented as points in a linear or affine feature space. Each image set is formulated as a convex geometric region. Geometric distance is used to measure similarity and make the final classification.

Image set classification is based on two parts: a model for the image sets, and a similarity metric to compare the representations. Our method uses a convolutional autoencoder to represent image sets. Autoencoders, which are often configured as deep networks, are used extensively to learn mappings for dimensionality reduction and compressed representations of images. They have grown in popularity in the last decade. A simpler and widely-used data dimensionality reduction technique is principal component analysis (PCA). PCA represents the data based on the orthogonal directions of maximum variance. PCA can give a poor representation for images with large number of features and low variance concentration [30]. The nonlinearity of neural networks on the other hand allows autoencoders to compress much more complex data while retaining information about the internal structure [25, 30].

2.4 Underwater Image Classification

Monitoring underwater organisms is crucial for better understanding of the ecosystem and affects of climate change. A lot of underwater image classification problems have similar challenges of variation in brightness, image quality and viewpoint orientations. There are recent works have tried to address these problems. Raitoharju et al. [38] have proposed a data enrichment algorithm to improve Neural Network based classification of aquatic macroinvertebrates. They created new images by rotations and mirroring of older images, which increases the dataset size, leading to better classification accuracy. Schoening et al. [40] propose an image patch based feature representation for the problem of seafloor classification. The paper from Chuang et al. [9] compared supervised and unsupervised feature extraction methods for fish species recognition. Their experiments show that unsupervised approach gives more accurate prediction of fish species. For a lot of underwater species recognition problem the choice of feature extraction and representation method is really crucial. In the next section, we would discuss the use of autoencoders for feature extraction from images.

2.5 Autoencoders

Autoencoders were initially proposed by Hinton et al. [25] and are frequently used for learning feature representation. Since then it has been used and studied for image representation, compression, and dimensionality reduction in wide range of data types. Liu et al. [32] proposed autoencoder features to predict well failures using SVM for final classification. The paper also compared the use of hand-crafted features with autoencoder features for classification. Bosch et al. [5] used LSTM units along with autoencoders to get features from time-series based educational data.

Most neural networks are trained to predict a target value or label Y given an input X, and a loss function is used to measure the difference between true and predicted labels. This loss is minimized over multiple iterations to increase the prediction accuracy. Autoencoders on the other hand use a combination of layers as an encoder to create a low dimensional representation and then use a decoder to reconstruct the input. The loss function is calculated using the difference between the input and reconstructed output data. The gradients are propagated through the decoder and the encoder network. Since the discovery of autoencoders there have been multiple variants of autoencoders that are applicable in wide range of problems. Vincent et al. [50] proposed denoising autoencoders which tries to reconstruct an image from a noised input image, thereby making the model robust to noise. Goroshin et al. [21] proposed an autoencoder architecture that limits the model's ability to reconstruct inputs which are not near the data manifold. The paper also shows that using different activation functions in the intermediate layers of autoencoder can be used to learn different features with interesting properties. Rifai et al. [39] adds a penalty term to the loss function which makes the model better at capturing the local directions of the data.

3 Methodology

Our prediction algorithm is based on two steps: feature set generation and classification. The training process involves two different models: an autoencoder trained to generate features from images, and a classifier trained to discriminate between invasive and non-invasive organisms.

3.1 Solution Description

Given an image set S (which typically contains different images of the same organism) with images $[x_1, x_2, \ldots, x_n]$ of size (a, b), our goal is to create a representation r of size z, where $z \ll a \cdot b$.

An autoencoder is used for feature extraction. Image x_i in the set has a corresponding feature f_i of size z, so there will be n features for n images in the set. Now these features are combined to create an average representation r of size z, where $r = \left(\sum_{i=1}^{n} f_i\right)/n$. The addition is done element-wise to create a final feature which is of the same size as the features from individual images. The autoencoder starts with an input size of $(a, b, 3)$ and the final layer of the encoder has an output size of z. Now, we should look at the details of the neural network architecture used for feature extraction.

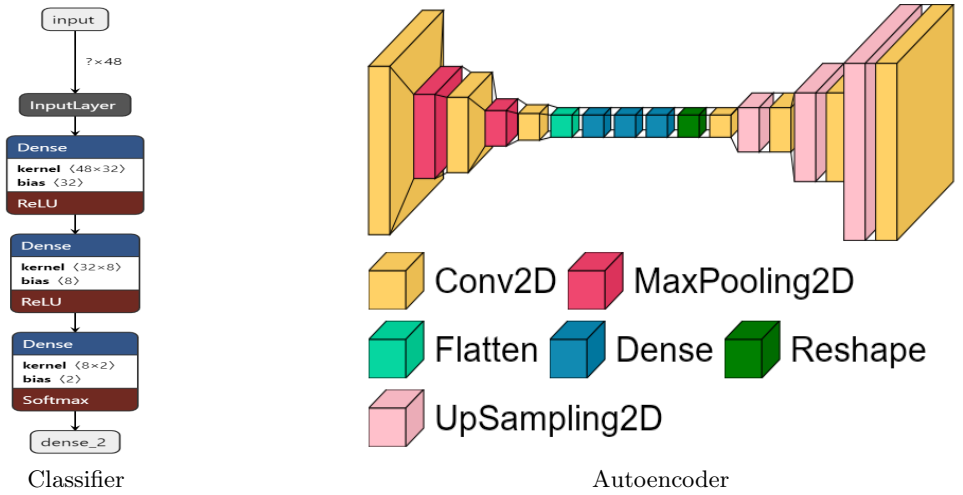
3.2 Convolutional Autoencoder

Let us assume we have images of size $(a, b, 3)$ given as input to the autoencoder. The autoencoder network $\Phi = \{\phi_e, \phi_d\}$ is formed of an encoder ϕ_e that creates a latent vector of size l_m and the decoder ϕ_d reconstructs the input image with the same size.

The network architecture is based on the VGG model [46], which generally means convolution layers of filter size 3×3, pooling layer of stride size 2×2 followed by dense layers with decreasing output size. The final layer of encoder is fully connected from the encoder to the decoder. This layer has an encoder output size of m equal to the size of the latent vector features l_m. The decoder reconstructs the image with a series of dense layers, convolution layers and up-sampling layers. For the decoder the pooling layer is replaced by up-sampling layer. We have used ReLU as activation in convolution layers and TanH in the final dense layer. The loss function is mean square error between the reconstructed image and input image. The parameters are learned using the Adam optimizer.

3.3 Classification Model

Once the autoencoder has been trained, it can be used to extract features from every image. These features are averaged with-in a set to give us a feature and label pair like (f_i, l_i) for organism i. Now, we have a neural network based classifier that is trained to predict the label from the features. The input features reduced by two dense layers with ReLU activation function. Let's assume that the input to the classifier are features like l_m of size m. So, this network is trained using three layers reducing the input from size m to the output size. The final layer has the output size of 2 with a softmax activation function. The loss function is categorical cross-entropy, the parameters are learned using the Adam optimizer and a dropout layer is used to regularize the network. Fig3. presents a detailed diagram of the classification model and the autoencoder model.



Classifier Autoencoder

Fig. 3. (Left) The classifier starts with features of size 48 and provides an output tensor of size 2, which is used to make the prediction. $? \times 48$ stand for the batch-size variable along with the input feature size. (Right) The autoencoder, with the input size of $(40, 40, 3)$. The output of the encoder is 48 features, which are used for feature averaging.

3.4 Activation Functions

For the autoencoder we use two types of activation functions. For the convolution layers we use ReLU activation, and for the encoder's output layer we use hyperbolic tangent TanH :

$$\text{TanH}(x) = \frac{(e^x - e^{-x})}{(e^x + e^{-x})} \tag{1}$$

Here, input $x \in \mathcal{R}$ and the output $\text{TanH}(x) \in [-1, 1]$. This means the autoencoder's output (latent) features have limited range, which acts as a regularization against extreme feature values [36].

Rectified Linear Unit (ReLU): The ReLU activation function is defined as $\text{ReLU}(x) = \max(0, x)$. This function eliminates negative values and eliminates the vanishing gradient problem observed with other activation function [36].

Softmax: The Softmax function is used to compute probability distribution from real valued vectors [36]. Softmax output tensors are in the range $[0, 1]$, with the sum of the tensors being equal to 1. For softmax activation the output tensor $f(x)$, given input tensor x of size k is derived as:

$$f(x)_i = \frac{\exp(x_i)}{\sum_{j=1}^{k} \exp(x_j)} \tag{2}$$

We used softmax activation at the final layer of classification and ReLU activation in the convolution layers of the autoencoder.

3.5 Loss Functions

The loss function for the autoencoder network is mean squared error (MSE). MSE in this case is the average of pixel-wise squared error between the input image and generated image.

$$\text{MSE}(Y, \hat{Y}) = \frac{1}{n}\sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2 \tag{3}$$

True Labels: Y and Predicted Labels: $\hat{Y}$, n is the number of pixel.

The classifier portion of the network uses a softmax activation function for the output layer along with categorical cross-entropy as the loss function. We use categorical encoding to encode the target label to numerical features with values between values of 0 to 1. Cross-entropy loss is computed from the sum of the negative logarithm of predictions made by the Neural Network. For our case with n samples and category $C = 2$, if ground truth is given by Y and prediction by $\hat{Y}$, where $Y, \hat{Y} \in [0, 1]$, the cross-entropy loss (CE) is given by:

$$\text{CE}(Y, \hat{Y}) = -\sum_{i=1}^{n}\sum_{c=1}^{C} Y_{ic} \cdot \log(\hat{Y}_{ic}) = -\sum_{i=1}^{n}(Y_{i1} \cdot \log(\hat{Y}_{i1}) + Y_{i2} \cdot \log(\hat{Y}_{i2})) \tag{4}$$

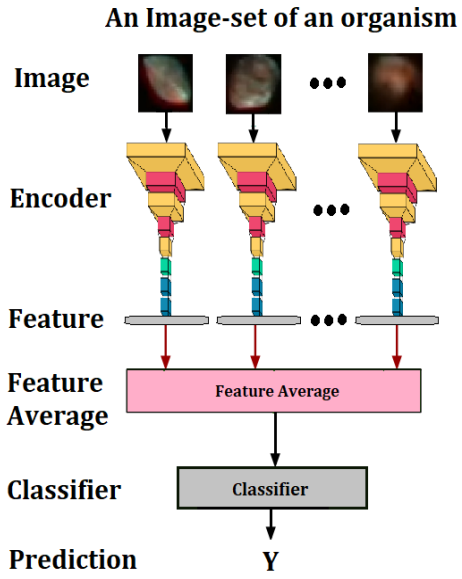


Fig. 4. Diagram of the feature averaging pipeline. The images in the set are from the same organism. Image features are extracted from all images using the encoder and the average feature is used for final classification.

3.6 Base Model

For comparison with neural network models we use a convolutional neural network classifier to classify individual images (as opposed to classifying a set of images per organism). The base model has two convolution layers of filter size 3×3 with a max pooling layer for each of them and one convolution layer of filter size 5×5. Finally there are the fully connected dense layers. The final layer of the network has softmax activation with categorical cross entropy as loss function. The weights are initialized using Xavier initializer and parameters are learned using Adam optimizer. We train the model for 20 epochs with a batch size of 32 and learning rate .001. A diagram of the base model with all the different layers is shown in Fig5. .

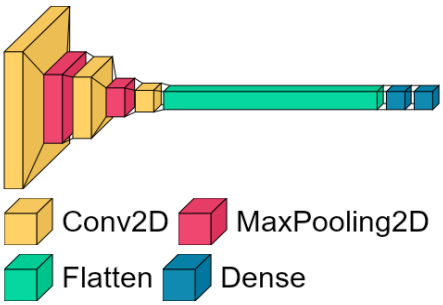


Fig. 5. This is the base neural network model which has an input of size $(40, 40, 3)$, three convolutional layers of filter size 3×3 and three dense layers.

3.7 Dataset

Our dataset contains a total of $4,374$ organisms with a total of $112,788$ images. There are 674 invasive organisms (quagga mussels) with $19,101$ images and $3,700$ non-invasive organisms with $93,687$ images. On average each organism has 25.78 instances captured from the video sample. The average image size is 22.56×19.46 pixels. We resize every image to a constant size of $40 \times 40 \times 3$ and use that as an input to the autoencoder. We trained two autoencoder models with latent feature size of 48 , 16 and 64. We trained classifier for each of those feature sets to separate the organisms into two categories: Invasive and Non-invasive.

Table 1. These are details about the neural network models used. The first row presents an autoencoder with an output vector of 64 features. The following two rows give details of the encoder and decoder which are used to construct that autoencoder. Next, we have the details of autoencoders with latent vectors of 48 and 16 features. This is followed by the classifiers and the base neural network model.

Model Type	# Parameters	# Convolution Layers	# Dense Layers
Autoencoder (64 features)	38,297	7	3
Encoder (64 features)	28,089	3	2
Decoder (64 features)	10,208	4	1
Autoencoder (48 features)	34,281	7	3
Encoder (48 features)	26,073	3	2
Decoder (48 features)	8,208	4	1
Autoencoder (16 features)	26,249	7	3
Encoder (16 features)	22,041	3	2
Decoder (16 features)	4,208	4	1
Classifier(64 features)	2,362	0	3
Classifier(48 features)	1,850	0	3
Classifier(16 features)	276	0	3
Base Model (CNN)	204,512	3	3

4 Results

4.1 Evaluation Metric

Our dataset has imbalance between different classes and the cost of a false negative (missing an invasive larvae) is potentially high. Therefore, accuracy alone might not be sufficient to evaluate the performance of the model. Thus we look at the following performance metrics:

Recall: Recall measures the percentage of the positive group that was correctly predicted to be positive by the model. Recall is not affected by imbalance because it is only dependent on the positive group. Recall does not consider the number of negative samples that are misclassified as positive, which can be problematic in problems containing class imbalanced data with many negative samples.

F1 Score: F1 score combines precision and recall using the harmonic mean, where coefficient β is used to adjust the relative importance of precision versus recall. The general formula with equal weight to precision and recall is given by:

$$\text{F1} = \frac{2 \cdot \text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}} = \frac{2TP}{2TP + FP + FN} \tag{5}$$

Balanced Accuracy: Balanced Accuracy is the average of the individual accuracy of all classes. Balanced accuracy is one of the most frequently used metrics when dealing with class imbalance.

$$\text{BAC} = \frac{1}{2}\left(\frac{TP}{TP + FN} + \frac{TN}{FP + TN}\right) \tag{6}$$

4.2 Quantitative Analysis

We split the dataset into train and test data for both the autoencoder and classifier. We use 80% for training and validation and 20% for testing. We shuffle the dataset before each training iteration to evaluate the models across the dataset. The shuffle is applied over organisms, so that the images from same organism are not in both traning and test set. We train the autoencoder for 20 epochs and the classifier for 200 epochs. We use the Xavier initializer to initiate the weights and use the Adam optimizer with a learning rate of .001 to learn the parameters. We visualize the training accuracy and loss against the number of training epochs. Then we show the comparative performance of our model against other popular machine learning methods.

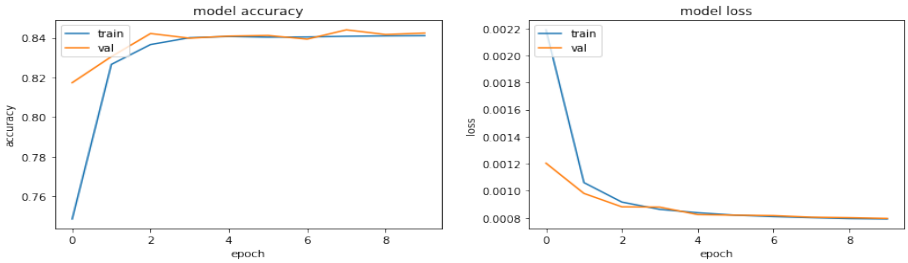


Fig. 6. (Left) Training accuracy of autoencoder against number of epochs. (Right) Autoencoder training loss against number of epochs. The terms 'train' and 'val' are for training and validation data.

4.3 Comparative Analysis

We shuffle the dataset before doing the train-test split for both the autoencoder and classifier. We perform 10 iterations of training and use the average score to compare the result. Other than neural network we also compare the results with other machine learning methods like a convolutional neural network, PCA + neural network, SVM classifier, and PCA + KNN.

We shuffled the data-set before each iteration of train-test split to validate the results across the dataset. For the base neural network we report the accuracy on individual images and also on organisms (image sets) based on majority vote. We have also reported the accuracy with under-sampling method, where we have used a subset of the non-invasive images to balance the two classes. For PCA we report accuracy on individual images and similar feature averaging for organisms-wise results. The accuracy for autoencoder based feature average is highest for the 48 features.

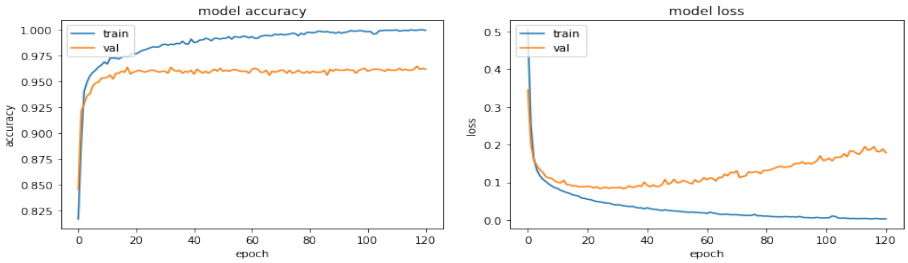


Fig. 7. (Left) Training accuracy of classifier against number of epochs. (Right) Classifier training loss against number of epochs. The terms 'train' and 'val' are for training and validation data. During the training we monitored validation loss for stop criteria .

Table 2. Experimental results on three evaluation metrics: F1-Score, Balanced Accuracy, and Recall. These results are based on the average of 10 experiments. The results at the top are based on classification of image-set of each organism (each prediction is for one organism using a set of images). The results at the bottom are based on individual images of all organisms in the test set (each prediction is for an individual image). The base neural network is trained to classify each image. For method 4 we used majority voting to make prediction on each organism. The results show that the feature averaging process gives reliable improvement.

Type	#	Method	Test Size	F1 Score	BAC	Recall
Classify Each Organism	1	Feature Averaging (64 features)	850	$97.1 \pm 0.9\%$	$98.2 \pm 0.7\%$	$96.3 \pm 0.5\%$
	2	Feature Averaging (48 features)	850	$97.1 \pm 0.3\%$	$98.2 \pm 0.3\%$	$96.3 \pm 0.4\%$
	3	Feature Averaging (16 features)	850	$90.5 \pm 0.3\%$	$95.2 \pm 1.2\%$	$88.8 \pm 1.5\%$
	4	Base Neural Network (CNN)	850	$88.1 \pm 0.7\%$	$89.4 \pm 0.3\%$	$82.5 \pm 0.6\%$
	5	PCA (Feature Average) + Neural Network	850	$86.7 \pm 0.6\%$	$92.5 \pm 0.7\%$	$85.5 \pm 0.4\%$
Classify Each Image	6	Base Neural Network (CNN)	20,196	$80.2 \pm 1.2\%$	$89.3 \pm 1.6\%$	$80.1 \pm 1.4\%$
	7	Base Neural Network (under-sampling)	20,196	$82.2 \pm 1.1\%$	$87.5 \pm 1.5\%$	$82.7 \pm 1.1\%$
	8	PCA + Neural Network	20,196	$66.8 \pm 1.1\%$	$82.9 \pm 1.3\%$	$54.9 \pm 0.9\%$
	9	SVM	20,196	$74.6 \pm 0.6\%$	$83.0 \pm 0.3\%$	$79.3 \pm 0.8\%$
	10	PCA+ 3-Nearest Neighbour	20,196	$64.2 \pm 10.0\%$	$78.7 \pm 5.0\%$	$68.1 \pm 9.0\%$

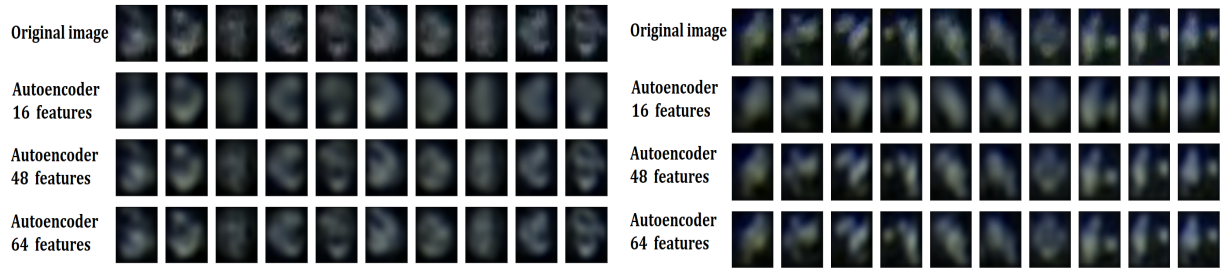


Fig. 8. Images of invasive and non-invasive species reconstructed by the autoencoders. Notice that larger autoencoder features create better reconstructions which also improves the accuracy of the final classification.

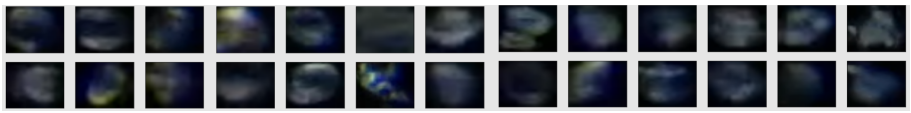


Fig. 9. These are some of the images incorrectly classified by the Base Neural Network (VGG). A lot of these images have low brightness or have different viewpoints that causes incorrect prediction. On, the other hand, Feature averaging on a set of images can often classify the organism correctly.

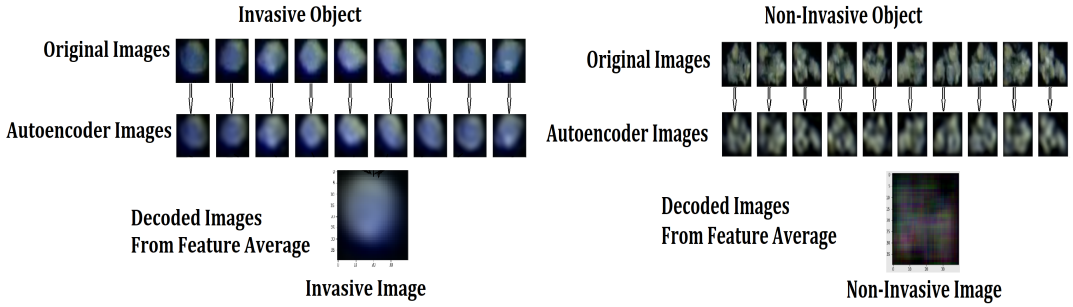


Fig. 10. 1. These are the invasive reconstruction created from images of an organism and at the bottom the decoder reconstruction from average representation. 2. Similar average representation of a Non-invasive organism

This is a binary classification problem. For single images, a convolutional neural network has F1 score of 80%. Instead of single image based classification, an image-set based classification can generally get higher accuracy in vast number of cases [42,47]. Our results show that autoencoder based feature averaging improves the accuracy significantly over single image classification and has consistent performance comparable to state-of the art image-set classification techniques. Moreover, it shows that, feature fusion applied over an image-set before classification has an advantage over voting at the end of classification.

5 Conclusion

This paper presents a framework to recognise invasive zebra and quagga musell larvae from videos of water samples. Our solution is based on image-set classification using a feature averaging method that creates a representation for each organism and the final prediction is made using the average representation. Our experiments establish the robustness of this method compared to other image classification techniques.

The spread of invasive species is a critical problem that has global impact. Our main contribution here is to show that it is possible to separate invasive species like zebra and quagga mussels from non-invasive species using deep learning based feature averaging technique. Our current classification process involves training two different neural networks: an autoencoder and a classifier based on a feed forward neural network. In future our goal is to be able to do the classification with an end-to-end neural network based model that can take advantage of the movement as well as shape of invasive species. Other than that, there are variables such as image/object size, weather, season (e.g. Fall or Spring), that might be relevant to the classification. Finally, there is opportunity to extend this work to include other invasive species that are relevant to United States water-bodies like Green crabs, Asian carp, hydrilla, Northern Snakehead etc.

References

1. Arizona game and fish: Invasive zebra mussels found in “moss ball” aquarium product. azgfd.com/invasive-zebra-mussels-found-in-moss-ball-aquarium-product-sold-at-aquarium-and-pet-supply-stores/
2. Arandjelovic, O., Shakhnarovich, G., Fisher, J., Cipolla, R., Darrell, T.: Face recognition with image sets using manifold density divergence. In: 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR’05). vol. 1, pp. 581–588. IEEE (2005)

3. Benton, G., Finzi, M., Izmailov, P., Wilson, A.G.: Learning invariances in neural networks. arXiv preprint arXiv:2010.11882 (2020)
4. Bochinski, E., Bacha, G., Eiselein, V., Walles, T.J., Nejstgaard, J.C., Sikora, T.: Deep active learning for in situ plankton classification. In: International Conference on Pattern Recognition. pp. 5–15. Springer (2018)
5. Bosch, N., Paquette, L.: Unsupervised deep autoencoders for feature extraction with educational data. In: Deep learning with educational data workshop at the 10th international conference on educational data mining (2017)
6. Cevikalp, H., Triggs, B.: Face recognition based on image sets. In: 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. pp. 2567–2573. IEEE (2010)
7. Chai, J., Zeng, H., Li, A., Ngai, E.W.: Deep learning in computer vision: A critical review of emerging techniques and application scenarios. Machine Learning with Applications **6**, 100134 (2021)
8. Chen, Y.C., Patel, V.M., Phillips, P.J., Chellappa, R.: Dictionary-based face recognition from video. In: European conference on computer vision. pp. 766–779. Springer (2012)
9. Chuang, M.C., Hwang, J.N., Williams, K.: Supervised and unsupervised feature extraction methods for underwater fish species recognition. In: 2014 ICPR Workshop on Computer Vision for Analysis of Underwater Imagery. pp. 33–40. IEEE (2014)
10. Conn, D., Lutz, R., Hu, Y.P., Kennedy, V.: Guide to the Identification of Larval and Postlarval Stages of Zebra Mussels, Dreissena spp. and the Dark False Mussel, Mytilopsis leucophaeata (January 1993)
11. Connelly, N.A., O'Neill, C.R., Knuth, B.A., Brown, T.L.: Economic impacts of zebra mussels on drinking water treatment and electric power generation facilities. Environmental management **40**(1), 105–112 (2007)
12. Cowart, D.A., Breedveld, K.G., Ellis, M.J., Hull, J.M., Larson, E.R.: Environmental DNA (eDNA) applications for the conservation of imperiled crayfish (decapoda: Astacidea) through monitoring of invasive species barriers and relocated populations. Journal of Crustacean Biology **38**(3), 257–266 (2018)
13. Daliri, M.R., Torre, V.: Robust symbolic representation for shape recognition and retrieval. Pattern recognition **41**(5), 1782–1798 (2008)
14. Deng, Y., Loy, C.C., Tang, X.: Image aesthetic assessment: An experimental survey. IEEE Signal Processing Magazine **34**, 80–106 (2017)
15. Durán, C., Lanao, M., Anadón, A., Touyá, V.: Management strategies for the zebra mussel invasion in the ebro river basin. Aquatic Invasions **5**(3), 309–16 (2010)
16. Fan, W., Yeung, D.Y.: Locally linear models on face appearance manifolds with application to dual-subspace based classification. In: 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06). vol. 2, pp. 1384–1390. IEEE (2006)
17. Feist, S.M., Lance, R.F.: Advanced molecular-based surveillance of quagga and zebra mussels: A review of environmental DNA/RNA (eDNA/eRNA) studies and considerations for future directions. NeoBiota **66**, 117 (2021)
18. Fitzgibbon, A.W., Zisserman, A.: Joint manifold distance: a new approach to appearance based clustering. In: 2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2003. Proceedings. vol. 1, pp. I–I. IEEE (2003)
19. Gas, P., Electric: PGE: Prevent the spread of quagga and zebra mussels. pge.com/en_US/about-pge/environment/what-we-are-doing/quagga-and-zebra-mussel-prevention-program/quagga-and-zebra-mussel-prevention-program.page
20. Gingera, T.D., Bajno, R., Docker, M.F., Reist, J.D.: Environmental DNA as a detection tool for zebra mussels dreissena polymorpha (Pallas, 1771) at the forefront of an invasion event in Lake Winnipeg, Manitoba, Canada. Management of Biological Invasions **8**(3), 287 (2017)
21. Goroshin, R., LeCun, Y.: Saturating auto-encoders. arXiv preprint arXiv:1301.3577 (2013)
22. Gutierrez, P., Cordier, A., Caldeira, T., Sautory, T.: Data augmentation and pre-trained networks for extremely low data regimes unsupervised visual inspection. In: Optical Metrology (2021)
23. Hadid, A., Pietikainen, M.: From still image to video-based face recognition: an experimental analysis. In: Sixth IEEE International Conference on Automatic Face and Gesture Recognition, 2004. Proceedings. pp. 813–818. IEEE (2004)
24. Hinton, G.E., Dayan, P., Revow, M.: Modeling the manifolds of images of handwritten digits. IEEE transactions on Neural Networks **8**(1), 65–74 (1997)
25. Hinton, G.E., Salakhutdinov, R.R.: Reducing the dimensionality of data with neural networks. Science **313**(5786), 504–507 (2006)
26. Johnson, L.E.: Enhanced early detection and enumeration of zebra mussel (dreissena spp.) veligers using cross-polarized light microscopy. Hydrobiologia **312**(2), 139–146 (1995)
27. Khashman, A.: Face recognition using neural networks and pattern averaging. In: ISNN (2006)
28. Khashman, A.: Intelligent face recognition: Local versus global pattern averaging. In: Australian Conference on Artificial Intelligence (2006)
29. Kim, T.K., Kittler, J., Cipolla, R.: Discriminative learning and recognition of image set classes using canonical correlations. IEEE Transactions on Pattern Analysis and Machine Intelligence **29**(6), 1005–1018 (2007)

30. Kovenko, V., Bogach, I.: A comprehensive study of autoencoders applications related to images. In: Proceeding of the International Conference on Information Technology and Interactions (IT&I-2020). pp. 43–54 (2020)
31. Lin, W., Hasenstab, K., Moura Cunha, G., Schwartzman, A.: Comparison of handcrafted features and convolutional neural networks for liver mr image adequacy assessment. Scientific Reports **10**(1), 1–11 (2020)
32. Liu, J., Jaiswal, A., Yao, K.T., Raghavendra, C.S.: Autoencoder-derived features as inputs to classification algorithms for predicting well failures. In: SPE Western Regional Meeting. OnePetro (2015)
33. Marshall, N.T., Stepien, C.A.: Invasion genetics from eDNA and thousands of larvae: A targeted metabarcoding assay that distinguishes species and population variation of zebra and quagga mussels. Ecology and evolution **9**(6), 3515–3538 (2019)
34. Martin Meder, M.J.: The effects of zebra mussel (dreissena polymorpha) infestations on property values: Evidence from Waupaca County, Wisconsin, USA. In: SSRN Electronic Journal (January 2014)
35. Massachusetts: Rapid response plan for the zebra mussel (dreissena polymorpha) in massachusetts. ENSR International (2005)
36. Nwankpa, C., Ijomah, W., Gachagan, A., Marshall, S.: Activation functions: Comparison of trends in practice and research for deep learning. arXiv preprint arXiv:1811.03378 (2018)
37. O'Neill, Jr., C.R., Dextrase, A.: The introduction and spread of the zebra mussel in north america. In: Proceedings of The Fourth International Zebra Mussel Conference (March 1994)
38. Raitoharju, J., Riabchenko, E., Meissner, K., Ahmad, I., Iosifidis, A., Gabbouj, M., Kiranyaz, S.: Data enrichment in fine-grained classification of aquatic macroinvertebrates. In: 2016 ICPR 2nd Workshop on Computer Vision for Analysis of Underwater Imagery (CVAUI). pp. 43–48. IEEE (2016)
39. Rifai, S., Vincent, P., Muller, X., Glorot, X., Bengio, Y.: Contractive auto-encoders: Explicit invariance during feature extraction. In: ICML (2011)
40. Schoening, T., Kuhn, T., Nattkemper, T.W.: Seabed classification using a bag-of-prototypes feature representation. In: 2014 ICPR Workshop on Computer Vision for Analysis of Underwater Imagery. pp. 17–24. IEEE (2014)
41. Shafait, F., Mian, A., Shortis, M., Ghanem, B., Culverhouse, P.F., Edgington, D., Cline, D., Ravanbakhsh, M., Seager, J., Harvey, E.S.: Fish identification from videos captured in uncontrolled underwater environments. ICES Journal of Marine Science **73**(10), 2737–2746 (2016)
42. Shah, S.A., Nadeem, U., Bennamoun, M., Sohel, F., Togneri, R.: Efficient image set classification using linear regression based image reconstruction. In: Proceedings of the IEEE conference on computer vision and pattern recognition workshops. pp. 99–108 (2017)
43. Shoeibi, A., Ghassemi, N., Alizadehsani, R., Rouhani, M., Hosseini-Nejad, H., Khosravi, A., Panahiazar, M., Nahavandi, S.: A comprehensive comparison of handcrafted features and convolutional autoencoders for epileptic seizures detection in eeg signals. Expert Systems with Applications **163**, 113788 (2021)
44. Shorten, C., Khoshgoftaar, T.M.: A survey on image data augmentation for deep learning. Journal of Big Data **6**, 1–48 (2019)
45. Siddiqui, S.A., Salman, A., Malik, M.I., Shafait, F., Mian, A., Shortis, M.R., Harvey, E.S.: Automatic fish species classification in underwater videos: exploiting pre-trained deep neural network models to compensate for limited labelled data. ICES Journal of Marine Science **75**(1), 374–389 (07 2017)
46. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014)
47. Sun, H., Zhen, X., Zheng, Y., Yang, G., Yin, Y., Li, S.: Learning deep match kernels for image-set classification. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 3307–3316 (2017)
48. Taylor, L., Nitschke, G.S.: Improving deep learning with generic data augmentation. 2018 IEEE Symposium Series on Computational Intelligence (SSCI) pp. 1542–1547 (2018)
49. Texas Parks and Wildlife Department: TPWD aquatic invasive species management FY 2020–2021. tpwd.texas.gov/landwater/water/aquatic-invasives/media/Statewide
50. Vincent, P., Larochelle, H., Bengio, Y., Manzagol, P.A.: Extracting and composing robust features with denoising autoencoders. In: Proceedings of the 25th international conference on Machine learning. pp. 1096–1103 (2008)
51. Wang, G., Hwang, J.N., Williams, K., Wallace, F., Rose, C.S.: Shrinking encoding with two-level codebook learning for fine-grained fish recognition. In: 2016 ICPR 2nd Workshop on Computer Vision for Analysis of Underwater Imagery (CVAUI). pp. 31–36 (2016)
52. Wang, R., Shan, S., Chen, X., Gao, W.: Manifold-manifold distance with application to face recognition based on image set. In: 2008 IEEE Conference on Computer Vision and Pattern Recognition. pp. 1–8. IEEE (2008)
53. Yamaguchi, O., Fukui, K., Maeda, K.i.: Face recognition using temporal image sequence. In: Proceedings Third IEEE International Conference on Automatic Face and Gesture Recognition. pp. 318–323. IEEE (1998)
54. Zhang, M., He, R., Cao, D., Sun, Z., Tan, T.: Simultaneous feature and sample reduction for image-set classification. In: AAAI (2016)
55. Zhao, Z.Q., Xu, S.T., Liu, D., Tian, W.D., Jiang, Z.D.: A review of image set classification. Neurocomputing **335**, 251–260 (2019)